



(12) **EUROPEAN PATENT APPLICATION**

(43) Date of publication:
21.09.2022 Bulletin 2022/38

(51) International Patent Classification (IPC):
G06K 9/00 ^(2022.01) **G06K 9/62** ^(2022.01)

(21) Application number: **21163101.5**

(52) Cooperative Patent Classification (CPC):
G06V 20/188; G06V 10/751; G06V 20/10

(22) Date of filing: **17.03.2021**

(84) Designated Contracting States:
AL AT BE BG CH CY CZ DE DK EE ES FI FR GB GR HR HU IE IS IT LI LT LU LV MC MK MT NL NO PL PT RO RS SE SI SK SM TR
Designated Extension States:
BA ME
Designated Validation States:
KH MA MD TN

(72) Inventors:
• **WAGNER, Andrew**
9052 Zwijnaarde (BE)
• **WAEGEMAN, Tim**
9052 Zwijnaarde (BE)

(74) Representative: **Rogiest, Wouter**
Patentales BV
Vennenbos 1A
9840 De Pinte (BE)

(71) Applicant: **Robovision**
9052 Zwijnaarde (BE)

(54) **IMPROVED VISUAL SERVOING**

(57) The present invention relates to a method for computing a pose for a robot head for handling an object by means of a handle connected to said object, comprising the steps of: (a) obtaining, by means of a vision sensor, an image of a scene comprising said object and said handle, said image comprising 3D information and preferably color information; (b) segmenting, by means of a trained segmentation NN, said image, according to a plurality of semantic components comprising at least a first

semantic component relating to said object and a second semantic component relating to said handle; (c) determining, based on said plurality of semantic components, handling data for handling said object, said handling data comprising a handling position being on said handle; and (d) computing, based on said handling data, a pose for said robot head, said pose comprising at least a robot head position for approaching said handle.

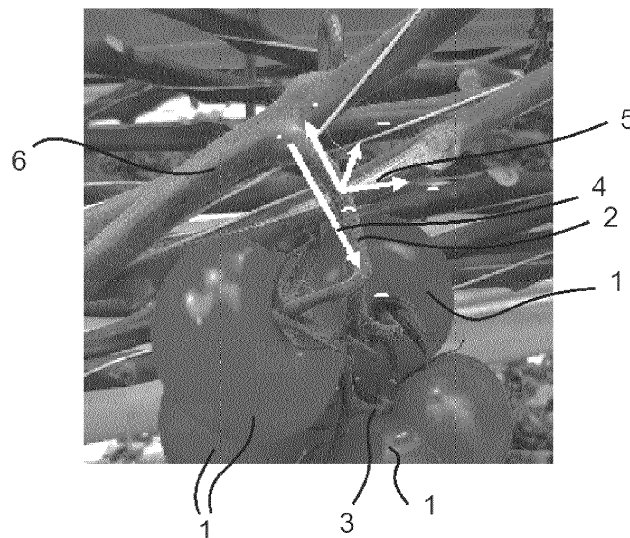


Fig. 1

Description**Field of the invention**

5 **[0001]** The present invention relates to handling of objects by means of robots based on deep learning and involving visual servoing.

Background art

10 **[0002]** In image analysis of 3D objects in the context of robot automation, visualization and 3D image reconstruction are fundamental for enabling accurate handling of physical objects.

15 **[0003]** Visual servoing is a method for robot control where camera or vision sensor input is processed to provide feedback for a robot control signal in a closed-loop manner. Finding suitable methods to continuously process visual inputs for robot control is a problem known to the skilled person, see, e.g., (Kragic, D, Christensen, HI, Survey on visual servoing for manipulation, Computational Vision and Active Perception Laboratory, 2002.

20 **[0004]** In the domain of visual servoing, (Quentin Bateux, Eric Marchand, Jürgen Leitner, François Chaumette, Peter Corke, Training DeepNeural Networks for Visual Servoing, IEEE International Conference on Robotics and Automation (ICRA), 2018) discloses the use of deep learning, yet is not adapted to the handling of objects.

25 **[0005]** US 2020/0008355 A1, CN 109948444 A, and WO 2020/155277 A1 disclose the use of deep learning but are not adapted for visual servoing and/or do not disclose any detail regarding how deep learning is to be applied.

30 **[0006]** JP 6749720 B1 discloses neural networks but does not disclose the use of neural networks for visual servoing

35 **[0007]** A known challenge of using deep learning for visual servoing is that typically a lot of data is required for training the system. Also, for an effective closed-loop control algorithm, the neural network needs to be processed sufficiently fast, as the latency will determine the operating speed.

40 **[0008]** The present invention aims at addressing the issues listed above.

Summary of the invention

45 **[0009]** According to a first aspect, the present invention provides a method for computing a pose for a robot head for handling an object by means of a handle connected to said object, said handle optionally being comprised in said object, comprising the steps of:

50 (a) obtaining, by means of a vision sensor, said vision sensor preferably mounted on said robot head, an image of a scene comprising said object and said handle, said image comprising 3D information and preferably color information;

55 (b) segmenting, by means of a trained segmentation NN, preferably a trained semantic segmentation NN, said image, according to a plurality of semantic components comprising at least a first semantic component relating to said object and a second semantic component relating to said handle;

60 (c) determining, based on said plurality of semantic components, handling data for handling said object, said handling data comprising a handling position being on said handle; and

65 (d) computing, based on said handling data, a pose for said robot head, said pose comprising at least a robot head position for approaching said handle.

70 **[0010]** A main advantage of such a method is the accurate and fast visual servoing provided by such a method. Particularly, the invention enables a visual servoing control loop with low latency. A detailed view of the object at several sides, as is often assumed in prior art methods, entails suboptimal handling in terms of speed. In order to actuate the robot head towards an object using visual servoing, the handle for the object is of prime interest.

75 **[0011]** In embodiments, the vision sensor is mounted on said robot head. This has the advantage of allowing a more accurate view on the object as the robot head approaches the object, according to several steps of the control loop.

80 **[0012]** In embodiments, the object belongs to a plurality of two or more objects comprised in said scene, and preferably the handle is shared by the plurality of objects being clustered objects.

85 **[0013]** In embodiments, the segmentation NN is a semantic segmentation NN. In embodiments, the segmentation NN is an instance segmentation NN.

90 **[0014]** In a second aspect, the invention provides a device handling an object, comprising a processor and memory comprising instructions which preferably, when executed by said processor, cause the device to execute a method

according to the invention.

[0015] In a further aspect, the invention provides a for handling an object, comprising:

- a robot head;
- a vision sensor, said vision sensor preferably mounted on said robot head;
- actuation means for actuating said robot head;
- a device, said device being connected to said vision sensor and said robot head, said device comprising a processor and memory comprising instructions which preferably, when executed by said processor, cause the device to execute a method according to the invention;

wherein said device is configured for:

- obtaining, from said vision sensor, an image of a scene comprising said object and a handle connected to said object, said image comprising 3D information and preferably color information;
- segmenting, by means of a trained segmentation NN, preferably a semantic segmentation NN, said image, according to a plurality of semantic components comprising at least a first semantic component relating to said object and a second semantic component relating to said handle;
- determining, based on said plurality of semantic components, handling data for handling said object, said handling data comprising a handling position being on said handle;
- computing, based on said handling data, a pose for said robot head, said pose comprising at least a robot head position for approaching said handle; and
- sending, to the actuation means, actuation instructions for actuating said robot head toward said robot head position;

wherein said vision sensor is configured for:

- acquiring said image;
- sending the image to said device;

wherein said actuation means is configured for:

- receiving actuation instructions from said device;
- actuating said robot head in accordance with said actuation instructions.

[0016] Preferred embodiments and their advantages are provided in the description and the dependent claims.

Brief description of the drawings

[0017] The present invention will be discussed in more detail below, with reference to the attached drawings.

Fig. 1 shows an example bunch of tomatoes to be detected, approached, and preferably picked.

Fig. 2 shows an example relating to clamping and cutting of a handle.

Fig. 3 illustrates an example of visual servoing cycles for picking and placing.

Description of embodiments

[0018] The present invention will be described with respect to particular embodiments and with reference to certain drawings but the invention is not limited thereto but only by the claims.

[0019] Furthermore, the various embodiments, although referred to as "preferred" are to be construed as exemplary manners in which the invention may be implemented rather than as limiting the scope of the invention.

[0020] The term "comprising", used in the claims, should not be interpreted as being restricted to the elements or steps listed thereafter; it does not exclude other elements or steps. It needs to be interpreted as specifying the presence of the stated features, integers, steps or components as referred to, but does not preclude the presence or addition of one or more other features, integers, steps or components, or groups thereof. Thus, the scope of the expression "a device comprising A and B" should not be limited to devices consisting only of components A and B, rather with respect to the present invention, the only enumerated components of the device are A and B, and further the claim should be interpreted as including equivalents of those components.

[0021] The terms "branch" and "stem" relate to embodiments wherein the object to be detected relates to a part of a plant, e.g., a fruit or a leaf. The terms "main stem" and "stem" are therein used in a relative manner, wherein the main stem branches out into one or more stems. Hence, the terms "main stem" and "stem" should not be construed as limiting, and merely relate to relative labels for respective parts of a plant.

[0022] The term "robot" refers to robot controllable for carrying out a movement. In embodiments the robot is a robot arm. Preferably the robot comprises a robot head at its distal end, wherein the vision sensor is preferably mounted on the robot head and/or mounted on a portion of the robot in the vicinity of the distal end. In embodiments the robot is suitable for performing pivoting and/or translation with respect to said head along at least one dimension, preferably at least two dimensions, more preferably three dimensions.

[0023] The term "image" relates to any representation of a generic scene, comprising visual data comprising any or any combination of pixels, voxels, vectors, and/or equivalent visual data. Any visual data in said image, e.g., a pixel or voxel, may be associated with one or more of color information, e.g. RGB information, and 3D information. In embodiments, the 3D information relates to depth data according to cartesian, cylindrical and/or spherical coordinates. In embodiments, the 3D information comprises, preferably consists of, depth information coded with one or more real value, e.g., one real value. In embodiments, the 3D information comprises, preferably consists of, information corresponding to two or more 2D sub-images relating to different viewing angles, e.g., a pair of a left sub-image and a right sub-image. In embodiments, the image is a voxel representation. In embodiments, the image is a pixel representation comprising, per pixel, RGBD data. In embodiments, the image comprises portions that are grayscale and/or portions that are colored, e.g., RGB-colored. In embodiments, the image is a grayscale image preferably comprising depth information. In embodiments, the image is a color image preferably comprising depth information.

[0024] In this document, the terms "object" and "handle" are generic terms referring to any generic object, wherein said handle is a second generic object that is directly or indirectly connected to said object and may serve as a handling means, e.g., a portion that can be clamped, with respect to said object. In this regard, the terms "object" and "handle" are merely relative functional descriptors that indicate a relation between the object and the handle. The terms cannot be construed as limiting the invention in anyway.

[0025] In this document, reference is made to "re-rendering". This relates to data for which depth information is available, e.g., RGBD data, which is different from an actual 3D voxel representation. By re-rendering based on the depth information, a partial re-rendering to 3D may be performed. However, for some portions of the scene, e.g., surfaces on the behind, it may not be possible to perform re-rendering.

[0026] In embodiments, the vision sensor relates to any of the following types 1-6:

	Type 1	Type 2	Type 3	Type 4	Type 5	Type 6
Technology	Stereo IR	Stereo IR	Structured light	Structured light	Lidar	Time of flight
Range	0.11m-10m	3m	0.3m-2m	0.16m-0.27m	0.25m-9m	0.25m-2.21m

[0027] A variety of such sensors is known to the skilled person. In embodiments, the vision sensor is based on stereo IR or structured light or visible light or lidar or time of flight or laser line scanning. In embodiments the range is between 1 mm and 3 m, preferably between 2 mm and 2 m, more preferable between 10 mm and 1 m. In embodiments, the vision sensor comprises an ASIC for minimal latency output. This has the advantage of increased speed for the overall visual servoing method. In embodiments, the vision sensor outputs RGB data output as well as depth information, abbreviated as RGBD. Depth information is preferably obtained from 3D reconstructions built into the sensor, based, e.g., on stereo IR and/or multiple cameras and/or multiple camera positions within the same vision sensor. In embodiments, the vision sensor is compact, with maximum dimension less than 300 mm, preferably less than 200 mm, and/or with weight less than 1 kg, preferably less than 0.5 kg, more preferably less than 0.3 kg. Preferably the vision sensor is comprised in a single housing so as to easily mount on the robot head. Preferably the vision sensor has latency less than 300 ms, more preferably less than 200 ms, even more preferably less than 100 ms, most preferably less than 20 ms. Preferably the vision sensor is suitable durable and/or moisture tolerant and/or able to be conveniently sterilized.

In embodiments, the vision sensor is able to provide a frame rate that is between 1 Hz and 100 Hz, e.g., 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 25, 30, 50 or 60 Hz.

[0028] In embodiments, the invention relates to a plurality of vision sensors comprising said vision sensor and a second vision sensor different from said vision sensor. Therein, each of the vision sensors may be comprised in the system according to the invention. The second vision sensor may be structurally different from said vision sensor, e.g. it may be of a different type of said types 1-6, but may also be structurally similar or equal, e.g., it may belong to the same type of said types 1-6. In embodiments with a plurality of vision sensors, said image of said scene comprising 3D information may be obtained from different respective images of the respective vision sensors. embodiments with a plurality of vision sensors, at least one, e.g., one, is not mounted on said robot head, and preferably at least one, e.g., one, is mounted on said robot head. The latter embodiments may be combined with embodiments wherein one or more vision sensors belonging to said plurality of vision sensors may be used primarily or solely for any or all of said steps (a) to (d), e.g. relating to approaching the object, whereas the remaining one or more vision sensors may be used primarily or solely in relation to the actual handling of the object, e.g., for clamping and/or cutting with respect to said object or handle.

[0029] In embodiments with a single visual sensor, the visual sensor may be a stationary visual sensor, or, equivalently, a static visual sensor, not moving along with the movement of the robot head. This may relate to visual servoing according to a static camera case, comparable to a human with human vision reaching to grab something without moving the head. In embodiments with a single visual sensor, the visual sensor may be a moving visual sensor actuated at least in part based on the movement of the robot head, e.g., by being mounted directly or indirectly on the robot head. Being mounted on the robot head may relate to "end of arm tooling" as known to the skilled person. This may relate to visual servoing according to a moving camera case, comparable to a dog catching a ball. In embodiments, a plurality of vision sensors is provided, e.g., two in number, wherein at least one, e.g., one, is a stationary vision sensor, and the other at least one, e.g., one, is a moving vision sensor.

[0030] In embodiments, the object comprises a rounded 3D surface corresponding to a distinctive feature on a depth image, such as a curvature. For instance, the curvature of a fruit or vegetable may be easily recognizable based on 3D features, and may be detected accordingly. In such embodiments, depth data helps identifying the object and segmenting the data. In embodiments, the object comprises a color that is distinctive, and the image comprises color information. For instance, colors in the red band of the spectrum are indicative of a tomato.

[0031] In some embodiments with a 2D approach, data is analyzed in a RGBD (Red Green and Blue plus Depth) representation, which is what many sensors with 3D information return. In some embodiments with a 3D approach, the RGBD image is converted to an unordered cloud of colored points (point cloud). In this representation, all three spatial dimensions may be handled uniformly, but the adjacency of pixels may be thrown out.

[0032] In embodiments, the 2D NN includes any or any combination of: U-net, U-net++ , see (Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, Jianming Liang, UNet++: A Nested U-Net Architecture for Medical Image Segmentation, 4th Deep Learning in Medical Image Analysis (DLMIA) Workshop, 2018.). In embodiments, the 3D NN includes any or any combination of Dynamic Graph Convolutional Networks (see Yue Wang, Yongbin Sun, Ziwei Liu, Sanjay E. Sarma, Michael M. Bronstein, Justin M. Solomon, Dynamic Graph CNN for Learning on Point Clouds, Computer Vision and Pattern Recognition, 2019), KPConv (Hugues Thomas, Charles R. Qi, Jean-Emmanuel Deschaud, Beatriz Marcotegui, Francois Goulette, Leonidas J. Guibas; The IEEE International Conference on Computer Vision (ICCV), 2019, pp. 6411-6420).

[0033] In preferred embodiments, the NN comprises a semantic segmentation NN being a 2D u-net. U-net is found to be particularly suitable due to increased speed and/or increased reliability, enabled by data augmentation and elastic deformation, as described in more detail in, e.g., (Ronneberger, Olaf; Fischer, Philipp; Brox, Thomas (2015). "U-Net: Convolutional Networks for Biomedical Image Segmentation. arXiv:1505.04597").

[0034] In preferred embodiments, said at least one trained 3D NN comprises a semantic segmentation NN being a 3D PointNet++. PointNet++ is an advantageous choice in that it provides both robustness and increased efficiency, which is enabled by considering neighbourhoods at multiple scales. More detail is provided, e.g., in (Charles R. Qi et al., PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, 2017, <https://arxiv.org/abs/1706.02413>).

[0035] Neural networks, NNs, need to be trained to learn the features that optimally represent the data. Such deep learning algorithms includes a multilayer, deep neural network that transforms input data (e.g. images) to outputs while learning higher level features. Successful neural network models for image analysis are semantic segmentation NNs. One example is the so-called convolutional neural network, CNN. CNNs contain many layers that transform their input using kernels, also known as convolution filters, consisting of a relatively small sized matrix. Other successful neural network models for image analysis are instance segmentation NNs. As known to the skilled person, instance segmentation NNs differ from semantic segmentation NNs in terms of algorithm and output, even in cases where the input, e.g. the images, are identical or very similar.

[0036] In general, semantic segmentation may relate, without being limited thereto, to detecting, for every pixel (in 2D) or voxel (in 3D), to which class of the object the pixel belong. Instance segmentation, on the other hand, may relate,

without being limited thereto, to detecting, for every pixel, a belonging instance of the object. It may detect each distinct object of interest in an image. In embodiments, 2D instance segmentation, preferably operating on 2D images, relates to Mask R-CNN, DeepMask, and/or TensorMask. In embodiments, 3D instance segmentation, preferably operating on a 3D point cloud generated from 2D images, relates to 3D-BoNet and/or ASIS.

[0037] In embodiments, the object belongs to a plurality of two or more objects comprised in said scene, and preferably the handle is shared by the plurality of objects being clustered objects. In embodiments, the object belongs to a plant and is comprised in a plurality of objects being a bunch. In embodiments, the handling of a plurality of objects relates to handling the objects at a shared handle for said objects, e.g., harvesting bunches of tomatoes. In embodiments, the handling of the plurality of objects relates to handling the objects by their respective handle, e.g. harvesting tomato by tomato or harvesting isolated fruits present in the same scene. In embodiments wherein the segmentation NN comprises an instance segmentation NN, the detection of instances may relate to identifying each instance of the plurality of objects being clustered objects, e.g. identifying the number of tomatoes in a bunch.

[0038] The term neural network, NN, refers to any neural network model. The NN may comprise any or any combination of a multilayer perceptron, MLP, a convolutional neural network, CNN, and a recurrent neural network, RNN. A trained NN relates to training data associated with a neural network based model.

[0039] In embodiments, said obtained image comprises color information, and said obtained image is a depth image comprising RGBD data. This has the advantage of being provided by many vision sensors, or, equivalently, visual sensors, available off the shelf while exhibiting low latency.

[0040] In embodiments, at least said determining of handling data comprises re-rendering a 3D image from said depth image.

[0041] In embodiments, said segmenting comprises 2D semantic segmentation performed on said depth image, wherein said trained semantic segmentation NN comprises a 2D NN, preferably a 2D u-net or a 2D rotation equivariant NN, being trained on a color representation comprising depth information as an artificial additional color.

[0042] In embodiments, said segmenting comprises re-rendering a 3D voxel representation from said depth image and performing 3D semantic segmentation on said 3D voxel representation, wherein said trained semantic segmentation NN comprises a 3D NN, preferably a PointNet++ or a 3D rotation equivariant NN.

[0043] In embodiments, the method comprises the further step of actuating said robot head toward said robot head position.

[0044] In embodiments, the method comprises, during or after actuating said robot head toward said new position, repeating at least one of step (a) to (d), preferably each of step (a) to (d), one or more times, preferably until a predetermined handling condition is met.

[0045] In embodiments, the pose further comprises a 3D approaching angle, wherein said computing comprises computing said approaching angle based on one or more of said plurality of semantic components for avoiding collision of said robot head with said one or more semantic components.

[0046] In embodiments, said handle extends between a distal end and a proximal end along a handle direction, wherein said determining of handling data comprises determining said handle direction belonging to said handling data, wherein the pose further comprises a 3D approaching angle, wherein said computing comprises computing said approaching angle based at least on said handle direction.

[0047] In embodiments, said robot head comprises clamping means for clamping said handle, wherein preferably said computed handling position and said approaching angle are directed at clamping and displacing said handle for separating said handle and said object from further portions of an entity, preferably a plant, to which the object and the handle belong; and/or wherein preferably the method comprises the further step of actuating said robot head toward said robot head position and actuating said clamping means for clamping and displacing said handle, and/or wherein preferably said robot head further comprises receiving means for receiving said object after said separating.

[0048] In embodiments, said robot head comprises clamping means for clamping said handle at said handling position preferably being a medial position. This may relate to grip optimization, wherein preferably the handling position is optimized for good grip. This may for instance relate to embodiments wherein a straight portion of the handle, e.g. a straight portion of a branch, is detected, and/or wherein a portion of the handle without protrusions is detected. In embodiments, the robot head comprises cutting means for cutting said handle at a cutting position preferably being a distal position. This may relate to further grip optimization, wherein preferably the cutting position is optimized, more preferably both the handling position and the cutting position are optimized, for good handling and cutting. This may again for instance relate to embodiments wherein a straight portion of the handle, e.g. a straight portion of a branch, is detected, and/or wherein a portion of the handle without protrusions is detected. In embodiments, the method preferably comprises the further step of computing, based on said second semantic component, said cutting position, and/or wherein preferably said computed handling position and said approaching angle are directed at clamping said handle at said handling position and cutting said handle at said cutting position for separating said handle and said object from further portions of an entity, preferably a plant, to which the object and the handle belong; and/or wherein preferably the method comprises the further step of actuating said robot head toward said robot head position and actuating said clamping

means for clamping said handle and actuating said cutting means for cutting said handle, and/or wherein preferably said robot head further comprises receiving means for receiving said object after said separating. In embodiments, which may optionally be combined with embodiments with a plurality of vision sensors comprising a stationary sensor, the method comprises the further step, after clamping, of verifying whether clamping was successful, preferably based on reiterating steps (a) to (d). This may have the advantage of detecting whether no parts of the scene, e.g. leaves, caused collision during approaching or clamping of the handle of the object, preferably before cutting, so as to ensure whether additional movement or repetition or other action is required before cutting. In embodiments, such detecting may advantageously be performed by a stationary vision sensor as such a vision sensor may provide for a better overview than a moving vision sensor. In embodiments, which may optionally be combined with embodiments with a plurality of vision sensors comprising a stationary sensor, the method comprises the further step, after cutting, of verifying whether cutting was successful, preferably based on reiterating steps (a) to (d).

[0049] In embodiments, said segmenting according to said plurality of semantic components relates to a third semantic component, wherein said object and said handle belong to a plant further comprising a main stem relating to said third semantic component, and wherein said computing of said pose relates to separating said object, preferably said object and said handle, from said third semantic component.

[0050] In embodiments, said robot head comprises cutting means, and wherein determining of said pose comprises

- finding pixels on a first boundary between the second segment component and the third segment component;
- clustering said pixels for determining a first position;
- finding a second position on a second boundary between the handle and any of: the object, or a fork connecting the object and the handle and being closest to said handle;
- determining a cutting position at between $1/4^{\text{th}}$ and $3/4^{\text{th}}$ between said first position and said second position, preferably at between $2/5^{\text{th}}$ and $3/5^{\text{th}}$ between said first position and said second position.

[0051] In embodiments, said handling position is determined as a point belonging to said handle being farthest removed from said object. This has the advantage of simplicity and ensures that the integrity of said object is maintained as much as possible.

[0052] In embodiments, said object relates to a plurality of clustered object instances, and wherein said handling position is determined as said point belonging to said handle being farthest removed from a center of said clustered object instances. This is advantageous as it enables handling complex objects comprising a plurality of clustered object instances, such as a bunch of tomatoes or a bunch of grapes.

[0053] In embodiments, the NN is rotation equivariant. In embodiments, the NN is translation and rotation equivariant.

[0054] The advantage of rotation equivariance may be understood from the following. A common challenge for machine learning on images is that a fully generic neural networks (i.e. "fully connected" or based on "multi-layer perceptrons") will only learn to recognize objects in the positions and rotations that they appear in the input images. There are two main approaches for handling this.

- A first, more "pragmatic" approach is to make sure that the objections of interest appear in all positions and orientations in the training dataset. This can be done either by increasing the amount of data collected, or by synthetically translating and rotating the captured inputs (and their corresponding labeled outputs). The latter approach is called "data augmentation". In embodiments, data augmentation is used.
- The second approach is the use of neural networks that are based on convolution. Convolution has the geometric property that if the input image is shifted spatially, the output is shifted by the same amount. This is called translation (or shift) equivariance. While the convolutional neural network architectures used in practice have accumulated some operators that compromise this equivariance (like max pooling), translation has contributed the boom in AI driven computer vision over the last decade. In embodiments, NN are used that are equivariant to both rotation and translation.

[0055] Rotation equivariance in deep learning have posed challenges when compared to translational equivariance, primarily because the group theory based mathematics necessary for doing a fully general and correct implementation are more complex.

[0056] Rotation equivariant NNs are known for specific applications, see, e.g., the "e2cnn" software library that makes experimentation with equivariant architectures feasible without a need to know group theory, see (Maurice Weiler, Gabriele Cesa, General E(2)-Equivariant Steerable CNNs, Conference on Neural Information Processing Systems (NeurIPS), 2019). This library defines rotation equivariant versions of the many of the same layers found in Tensorflow and in pytorch.

[0057] Applicant has found such rotation equivariant NNs to be particularly useful for visual servoing, as distinguished

from other problems for which a rotation equivariance NN may be less useful. In many applications, the objects of interest do indeed always appear in the same orientation in the image. For example, in street scenes, pedestrians and cars are usually not "upside down" in the image. However, in visual servoing applications, the vision sensor is mounted on the robot head, and will not always be upright; it will rotate as necessary to align with the object, and the object appears in a variety of orientations.

[0058] In the context of the invention, visual servoing may relate to any automation wherein the vision system is in the control loop. This may relate to any moving or stationary vision sensor. A moving sensor may have the advantage of getting a better view while approaching an object. On the other hand, a stationary sensor may have many advantages related to detection of touch by accident, occlusions, oversight of the detection of both the action and the effect of an action. In this regard, a stationary sensor may advantageously provide a supervisor concept either by itself (as single sensor) or as complementing a moving visual sensor. Furthermore, having only a stationary vision sensor may provide faster execution of detection and actuation, and may reduce the number of iterations in the control loop.

[0059] In embodiments with a 2D rotation equivariance NN, U-Net-like architectures are preferred, preferably based on rotation equivariant operators from (Maurice Weiler, Gabriele Cesa, General E(2)-Equivariant Steerable CNNs, Conference on Neural Information Processing Systems (NeurIPS), 2019). In embodiments with a 2D NN, Furthermore, some of the translational equivariance that is lost in typical naive max pooling downsampling implementations is recovered based on the method disclosed in (Richard Zhang. Making Convolutional Networks Shift-Invariant Again, International Conference on Machine Learning, 2019).

[0060] In embodiments, the NN involves only equivariant layers. In embodiments, the NN involves only data augmentation. In embodiments, the NN involves both equivariant layers and data augmentation.

[0061] In embodiments with a 3D rotation equivariance NN, the NN preferably comprises one or more neural network architectures based on the "e3cnn" library, see (Mario Geiger et al, (2020, March 22). github.com/e3nn/e3nn (Version v0.3-alpha). Zenodo. doi:10.5281/zenodo.3723557). Applicant has found this to be particularly advantageous. Indeed, for data in a 3D point cloud representation, the motivation for equivariance is even stronger than in 2D. While a 2D network can at best be equivariant to rotations about the viewing axis, a 3D network can be equivariant to any 3D rotation. The "e3cnn" library, like the "e2nn" library, contains definitions for convolutional layers that are both rotation and translation equivariant.

[0062] It is noted that the space of 3D rotation equivariance NNs is even less explored than that of 2D rotation equivariance NN.

[0063] In embodiments, the NN involves only equivariant layers. In embodiments, the NN involves only data augmentation. In embodiments, the NN involves both equivariant layers and data augmentation.

[0064] In embodiments, said semantic segmentation NN is a CNN. In embodiments, the NN comprises any or any combination of: 2D u-net, 3D u-net, Dynamic Graph CNN (DGCNN), PointNet++. In preferred embodiments, semantic segmentation in two dimensions is done with a convolutional neural network, CNN. In alternative embodiments, instead of a 2D CNN, also a 2D NN that is not convolutional may be considered. In preferred embodiments, segmentation in three dimensions is done with a neural network that may either be convolutional, such as a DGCNN, or non-convolutional, such as PointNet++. In embodiments, another variant of PointNet++ relating to PointNet may be considered without altering the scope of the invention. In preferred embodiments, semantic segmentation with a 2D CNN relates to u-net. In preferred embodiments, semantic segmentation with a 3D NN relates to DGCNN or PointNet++. Herein, DGCNN may relate to methods and systems described in (Yue Wang et al., Dynamic Graph CNN for Learning on Point Clouds, CoRR, 2018, <http://arxiv.org/abs/1801.07829>), and PointNet++ may relate to methods and systems described in (Charles R. Qi et al., PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, 2017, <https://arxiv.org/abs/1706.02413>).

[0065] In embodiments, said actuation relates to actuating said robot head and preferably furthermore comprises actuating said actuation means and/or said cutting means.

[0066] The invention involves obtaining an image preferably comprising color information and 3D information.

[0067] In embodiments, the robot head comprises clamping means which may be used for applications of gripping objects, removing objects from belt conveyors or baskets, transportation of objects and assortment of objects. However, other tasks could be handled, as well. Examples of objects that are gripped by the robotic element include industrial products, packaged goods, food, entire plants, and material such as metal or woods. However, organisms such as crops or fishery can be handled, as well. Thus, the objects that are handled are not limited to objects of a specific category.

[0068] In embodiments, the robot head may comprise cutting means. Robot heads of different shapes or different types can be used in embodiments according to the invention.

[0069] Below, the invention is illustrated according to a plurality of example embodiments, which are not intended to limit the scope of the invention in any way.

Example 1: example method with tomato bunch

[0070] In example embodiments, the object is a tomato (1) indirectly connected to the handle (2). Particularly, in example embodiments the tomato (1) is an object belonging to a plurality of two or more clustered objects (1) being a bunch of tomatoes, and the handle is the branch connecting the bunch of tomatoes to the main stem, also referred to as peduncle (2). Thereby, in example embodiments, said peduncle (2) and said tomato (1) are connected by a fork, also referred to as a pedicel (3). The handle is to be clamped and cut in order to pick the bunch of tomatoes.

[0071] Fig. 1 shows an example bunch of tomatoes (1) to be detected, approached, and preferably picked. Via a pedicel (3), each tomato (1) is connected to the handle being the peduncle (2), which in turn is connected to the main stem, or for short, the stem (6).

[0072] This example further considers choosing the handling pose of the robot, in this case comprising both clamping and cutting, preferably based on a segmented depth image.

[0073] Preferably, the method is based on the pixel wise segmentation of the depth image into different classes (i.e. tomato fruit, main stem, stem cutting point candidates) as input, and comprises computing one 6DOF pose that the robot should move to in order to cut the fruit as output. Hereby, 6DOF relates to six degrees of freedom, i.e. three coordinates, e.g. xyz coordinates, and a 3D approaching angle, e.g. alpha, beta, gamma. In preferred embodiments, the involved NN is trained according to manual labeling, which may relate to labeled ground truth segmentations. In embodiments, the NN is a 2D network. In other embodiments, the NN is a 3D network.

[0074] This example furthermore relates to closed loop integration testing. Such testing may relate to one of, or both of, a "tabletop" scenario and the "greenhouse" scenario. Example embodiments relate to both of the two scenarios yet may be focused on one of the two to provide additional insight.

[0075] The tabletop scenario may relate to a simplified dataset in a lab setting, with tomatoes put on a tabletop for carrying out the invention, preferably including training of any NN involved. The greenhouse scenario may relate to an industrial greenhouse setting as known to the skilled person, wherein the invention is carried out, preferably including training of any NN involved. In embodiments, the method is carried out in any of both scenarios without requiring new training. This said, the greenhouse scenario may relate to more realistic lighting and/or contending with varying amounts of natural and artificial lighting of various colors and/or increased issues of reachability and visibility and/or foliage or other stems occluding the fruit and/or other plants in the background.

[0076] In example embodiments relating to the "tabletop" scenario, while preferably also being applicable to the greenhouse scenario, the method comprises the steps of:

- computing the center of the bunch of fruit being the bunch of objects;
- choosing the farthest point labeled as "stem" being the handle as the approach point being the handling position;
- determining the 3D direction of the stem where it was cut;
- computing the 6DOF pose that is aligned with the stem and closest to the current gripper pose.

[0077] In example embodiments, the method should first find the merge point of the tomato bunch stem to the main stem, and then estimate the 3D pose of the cutting point. Such methods may relate to the "greenhouse" scenario, while preferably also being applicable to the tabletop scenario, the method comprises the steps of:

- Find pixels that are on a boundary between parts of the data labeled "main stem", i.e. corresponding to the third semantic component, and the parts labeled as "side branches", i.e. handles corresponding to the second semantic components, and cluster them. This may advantageously identify locations where the main stem meets a side branch.
- For each side branch, extract the connected component for the branch. Find the first fork in that branch (going to the first fruit).
- Find a point midway between the above two points. This will be an approximate cutting point.
- The vector (3) between the points, i.e. the handle direction, preferably computed in the first two steps of a cycle, e.g., a cycle according to Example 3, runs along the stem. Hence, an additional vector to fully define the cutting pose is determined. In embodiments this is chosen to be the vector closest to the "down" direction, as ascertained using knowledge that the robot is mounted horizontally.
- Choose a third vector to complete the basis for the rotational part of the coordinate frame.

[0078] The NN may be any of a u-net, or PointNet++, a rotation equivariant NN such as the one disclosed in (Maurice Weiler, Gabriele Cesa, General E(2)-Equivariant Steerable CNNs, Conference on Neural Information Processing Systems (NeurIPS), 2019).

5 Example 2: example relating to clamping and cutting

[0079] Fig. 2 shows an example relating to clamping and cutting of a handle (2) connected on one end to an object (not shown) and connected on the other end to a further portion (6) of an entity to which the handle and the object belong. As the object is not shown, it is also noted that the figure is not drawn to scale. In examples, this may relate to the object being a tomato (1), the handle (2) being the peduncle (2), and the further portion being the main stem (6). In such applications, it is very important that the main stem is never cut, and the cutting of the handle generally has to be done with high precision, e.g., with handle length less than 100 mm or even less than 50 mm.

[0080] The object may correspond to a first semantic component, the further portion (6) to a third semantic component, and the handle (2) to a second semantic component. The handle is to be clamped and cut in order to separate the object from the further portion.

[0081] The robot head comprises clamping means for clamping (21) said handle (2) at said handling position, preferably being a medial position (21a), and cutting means for cutting said handle (2) at a cutting position preferably being a distal position (22a). The method comprises the further step of actuating said robot head toward said robot head position and actuating said clamping means for clamping said handle and actuating said cutting means for cutting said handle. The method comprises the further step of computing, based on said second semantic component, said cutting position, wherein said computed handling position and said approaching angle are directed at clamping said handle at said handling position and cutting said handle at said cutting position for separating said handle and said object from further portions of an entity, preferably a plant, to which the object and the handle belong. Preferably, after cutting at the cutting position, the handle is cut again at a second cutting position (22b) while still being clamped by the clamping means. This may yield a better finishing of the object, wherein the remains of the handle is smaller, leading to a more compact object, and/or wherein the end of the remaining part of the handle is cut more evenly, providing for better finishing of the object after cutting. The said robot head further comprises receiving means for receiving said object after said separating. Particularly, the receiving means may receive the object after the handle is cut at the second cutting position (22b).

[0082] In examples, the method may relate to

- finding pixels on a first boundary between the second segment component and the third segment component;
- clustering said pixels for determining a first position;
- finding a second position on a second boundary between the handle (2) and any of: the object (1), or a fork (3) connecting the object (1) and the handle (2) and being closest to said handle (2);
- determining a cutting position (22a) at between $1/4^{\text{th}}$ and $3/4^{\text{th}}$ between said first position and said second position, preferably at between $2/5^{\text{th}}$ and $3/5^{\text{th}}$ between said first position and said second position, most preferably halfway between said first position and said second position.

Example 3: example relating to cycles

[0083] In this example, visual servoing is applied to the problem of picking and placing of a single tomato bunch. Fig. 3 illustrates an example of visual servoing cycles for picking and placing.

[0084] The visual servoing example relates to time budget with target times for each computation that must be performed. A proposed cycle time budget for the act of harvesting a single tomato bunch can be seen in Fig. 3. Each row of arrows is a subdivision of the time from the higher level task in the row above it.

[0085] The first row shows the picking phase (30) with a time budget of 1.2 s, and the placing phase (40), with a time budget of 1 s.

[0086] The second row shows a visual control cycle (31a) with a time budget of 250 ms, followed by three more visual control cycles (31 b-d). This is continued with the closing of the clamping means (32), or, equivalently the gripper, with a time budget of 200 ms, ending the picking phase. This is followed by the move to the place point (41), with a time budget of 400 ms, the clamping means release (42), with a time budget of 200 ms, and the move to home (43), with a time budget of 300 ms.

[0087] The third row shows a single step (301a) of the visual servoing routine, with a time budget of 31 ms. The step is repeated seven more times (301b-h).

[0088] The fourth row shows the phases of the (first) single step (301a), i.e. the view prediction (3001a), relating to

obtaining the image and segmenting it, the stem detection (3002a), relating to determining handling data, and the cut pose computing (3003a).

[0089] In example embodiments, operation is provided with a 4 Hz visual servoing control update frequency. This gives a cycle time of 250 ms for all analyses performed in the control loop. For example, in one control time slice 8 hypotheses may be required by the view selection / path planning algorithm. This consumes half of the cycle time, the cutting point analysis code must run at $2 * 8 * 4$ Hz, with time to spare for the view selection code. This gives an analysis time for each simulated view of approx. 15 ms. To enable this time frame, in preferred embodiments, scene analysis is performed within a 10 ms time budget.

Example 4: examples relating to picking

[0090] In example embodiments of this example, separating the object from further portions of an entity to which the object belongs relates to picking.

[0091] In example embodiments, the object is a tomato. This may relate to Example 1.

[0092] In example embodiments the object is a grape belonging to a plurality of two or more clustered objects being a bunch of grapes. The handle is shared by the plurality of objects and is to be clamped and cut in order to pick the bunch of grapes.

[0093] In example embodiments, the object is a leaf of a plant, preferably an old leaf that is to be removed in order to improve yield of the plant, e.g., a tomato plant, and/or any leaf of any plant that requires partial or full deleafing. In such embodiments, the leaf, e.g., corresponding to a first semantic component, is connected to the main stem, e.g., corresponding to a third semantic component, through a handle being a petiole, e.g., corresponding to a second semantic component. In example embodiments, the handle is to be clamped and displaced in order to pick the leaf. In other example embodiments, relating to the same or to other plants, the handle is to be clamped and cut in order to pick the leaf.

[0094] In example embodiments the object is a cucumber extending between a free end and a plant-related end, and the handle is the plant portion connecting the cucumber at the plant-related end to further portions of the plant. The handle is to be cut in order to pick the cucumber.

[0095] In example embodiments the object is an apple or a pear, and the handle is the stalk and/or pedicel. In example embodiments, the handle is to be clamped and displaced in order to pick the apple or pear, wherein the displacing may or may not relate to a twisting motion. In alternative embodiments, the handle may also be clamped and cut in order to pick the apple or pear.

Example 5: examples relating to displacing

[0096] In example embodiments of this example, separating the object from further portions of an entity to which the object belongs relates to displacing.

[0097] In example embodiments, the object is a device part belonging to an entity being an electronic device, e.g., a modem or a computer, that requires dismantling. In such embodiments, one or more object types may be predetermined as being recognizable, preferably distinguishable, device parts for which the neural network is trained. Thereby, for each object type, a segmentation according to object and handle is performed, e.g., wherein one or more portions of the object are identified as advantageous, e.g., safe, positions for handling the object, corresponding to a second semantic component, and wherein, e.g., the remaining portions of the object correspond to the first semantic component. Thereby, separating the object from the electronic device relates to clamping the object at its handle and displacing it so as to remove it from further portions of the electronic device. In such example embodiments, the robot may be further configured to sort said device parts.

[0098] In example embodiments, the object is a loose object gathered together with other loose objects in an organized or disorganized fashion in a common container. Herein, the whole of container and loose objects is the entity to which the object belongs. Thereby, no cutting is involved, and the displacing of the handle merely relates to separating the object from the common container, without any physical connection between the handle and the further portions of the entity to which the object belongs. In such embodiments, one or more object types may be predetermined as being recognizable, preferably distinguishable, device parts for which the neural network is trained. Thereby, for each object type, a segmentation according to object and handle is performed, e.g., wherein one or more portions of the object are identified as advantageous, e.g., safe, positions for handling the object, corresponding to a second semantic component, and wherein, e.g., the remaining portions of the object correspond to the first semantic component. Thereby, separating the object from the container relates to clamping the object at its handle and displacing it so as to remove it from the container. In some example embodiments, the robot may be further configured to label the object with a sticker according to its object type and/or according to other information determined through an additional information determining step. In further examples, the robot may be further configured to then sort the objects with sticker and/or to then put the objects with sticker in a second common container. In other example embodiments, the robot may be further configured to,

based upon a 3D feature of the object and/or the object type of the object, select one or more additional objects so as to obtain a predetermined selection of objects that is separated from the container.

5 Claims

1. Method for computing a pose for a robot head for handling an object (1) by means of a handle (2) connected to said object (1), comprising the steps of:

- (a) obtaining (3001), by means of a vision sensor, said vision sensor preferably mounted on said robot head, an image of a scene comprising said object (1) and said handle (2), said image comprising 3D information and preferably color information;
- (b) segmenting (3001), by means of a trained segmentation NN, preferably a semantic segmentation NN, said image, according to a plurality of semantic components comprising at least a first semantic component relating to said object (1) and a second semantic component relating to said handle (2);
- (c) determining (3002), based on said plurality of semantic components, handling data for handling said object, said handling data comprising a handling position (21a) being on said handle (2); and
- (d) computing (3003), based on said handling data, a pose for said robot head, said pose comprising at least a robot head position for approaching said handle (2),

wherein preferably the object belongs to a plurality of two or more objects comprised in said scene, and wherein more preferably the handle is shared by the plurality of objects being clustered objects.

2. Method of claim 1, wherein said vision sensor is mounted on said robot head.

3. Method of claim 1 or 2, wherein said obtained image comprises color information, and wherein said obtained image is a depth image comprising RGBD data.

4. Method of claim 3, wherein at least said determining (3002) of handling data comprises re-rendering a 3D image from said depth image.

5. Method of claims 1-4, wherein said segmenting (3001) comprises 2D semantic segmentation performed on said depth image, wherein said trained semantic segmentation NN comprises a 2D NN, preferably a 2D u-net or a 2D rotation equivariant NN, being trained on a color representation comprising depth information as an artificial additional color.

6. Method of claims 1-5, wherein said segmenting (3001) comprises re-rendering a 3D voxel representation from said depth image and performing 3D semantic segmentation on said 3D voxel representation, wherein said trained semantic segmentation NN comprises a 3D NN, preferably a PointNet++ or a 3D rotation equivariant NN.

7. Method of claims 1-6, wherein the method comprises the further step of actuating said robot head toward said robot head position, and wherein preferably the method comprises, during or after actuating said robot head toward said new position, repeating step (a) to (d) one or more times, until a predetermined handling condition is met.

8. Method of claims 1-7, wherein the pose further comprises a 3D approaching angle, wherein said computing (3003) comprises computing said approaching angle based on one or more of said plurality of semantic components for avoiding collision of said robot head with said one or more semantic components.

9. Method of claims 1-8, wherein said handle (2) extends between a distal end and a proximal end along a handle direction (4), wherein said determining (3002a) of handling data comprises determining said handle direction (4) belonging to said handling data, wherein the pose further comprises a 3D approaching angle, wherein said computing (3003) comprises computing said approaching angle based at least on said handle direction.

10. Method of claims 1-9, wherein said robot head comprises clamping means for clamping said handle (2), wherein said computed handling position and said approaching angle are directed at clamping and displacing said handle for separating said handle and said object from further portions of an entity, preferably a plant, to which the object and the handle belong; wherein preferably the method comprises the further step of actuating said robot head toward said robot head position and actuating said clamping means for clamping and displacing said handle, and wherein

preferably said robot head further comprises receiving means for receiving said object after said separating.

11. Method of claims 1-9, wherein said robot head comprises clamping means for clamping (21) said handle (2) at said handling position preferably being a medial position (21a), and cutting means for cutting said handle (2) at a cutting position preferably being a distal position (22a), wherein the method comprises the further step of computing, based on said second semantic component, said cutting position, wherein said computed handling position and said approaching angle are directed at clamping said handle at said handling position and cutting said handle at said cutting position for separating said handle and said object from further portions of an entity, preferably a plant, to which the object and the handle belong; wherein preferably the method comprises the further step of actuating said robot head toward said robot head position and actuating said clamping means for clamping said handle and actuating said cutting means for cutting said handle, and wherein preferably said robot head further comprises receiving means for receiving said object after said separating.

12. Method of claims 1-10, wherein said segmenting (3001) according to said plurality of semantic components relates to a third semantic component, wherein said object (1) and said handle (2) belong to a plant further comprising a main stem (6) relating to said third semantic component, and wherein said computing (3003) of said pose relates to separating said object (1), preferably said object (1) and said handle (2), from said third semantic component.

13. Method of claim 12, wherein said robot head comprises cutting means, and wherein determining of said pose comprises

- finding pixels on a first boundary between the second segment component and the third segment component;
- clustering said pixels for determining a first position;
- finding a second position on a second boundary between the handle (2) and any of: the object (1), or a fork (3) connecting the object (1) and the handle (2) and being closest to said handle (2);
- determining a cutting position at between 1/4th and 3/4th between said first position and said second position, preferably at between 2/5th and 3/5th between said first position and said second position.

14. Method of claims 1-13, wherein the NN is rotation equivariant.

15. Device for handling an object (1), comprising a processor and memory comprising instructions which preferably, when executed by said processor, cause the device to execute a method according to claims 1-14.

16. System for handling an object (1), comprising:

- a robot head;
- a vision sensor, said vision sensor preferably mounted on said robot head;
- actuation means for actuating said robot head;
- a device, preferably the device according to claim 15, said device being connected to said vision sensor and said robot head, said device comprising a processor and memory comprising instructions which preferably, when executed by said processor, cause the device to execute a method according to claims 1-14;

wherein said device is configured for:

- obtaining (3001), from said vision sensor, an image of a scene comprising said object (1) and a handle (2) connected to said object (1), said image comprising 3D information and preferably color information;
- segmenting (3001), by means of a trained segmentation NN, preferably a trained semantic segmentation NN, said image, according to a plurality of semantic components comprising at least a first semantic component relating to said object (1) and a second semantic component relating to said handle (2);
- determining (3002), based on said plurality of semantic components, handling data for handling said object, said handling data comprising a handling position (21a) being on said handle (2);
- computing (3003), based on said handling data, a pose for said robot head, said pose comprising at least a robot head position for approaching said handle (2); and
- sending, to the actuation means, actuation instructions for actuating said robot head toward said robot head position;

wherein said vision sensor is configured for:

- acquiring said image;
- sending the image to said device;

wherein said actuation means is configured for:

5

- receiving actuation instructions from said device;
- actuating said robot head in accordance with said actuation instructions,

10

wherein preferably the object belongs to a plurality of two or more objects comprised in said scene, and wherein more preferably the handle is shared by the plurality of objects being clustered objects.

15

20

25

30

35

40

45

50

55

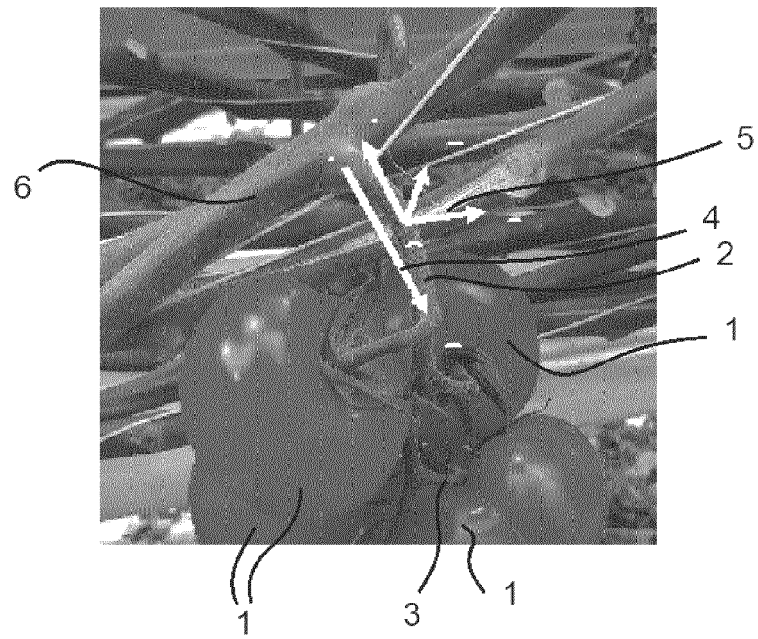


Fig. 1

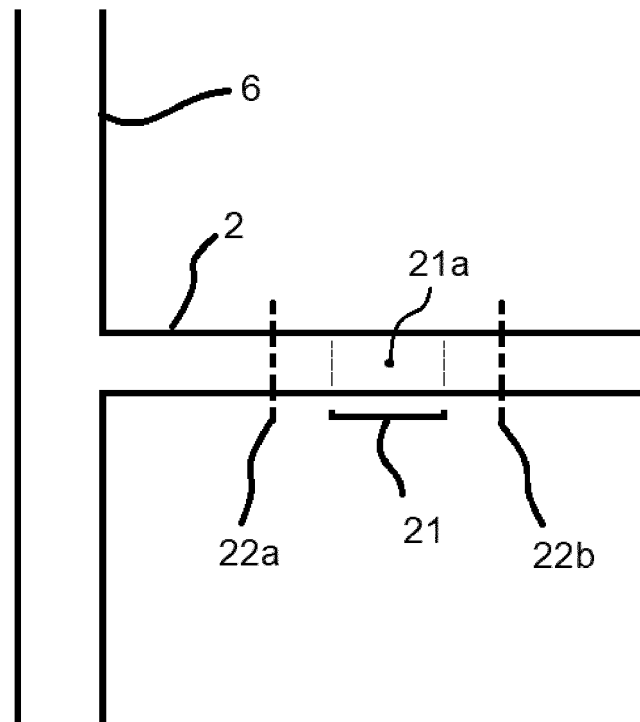


Fig. 2

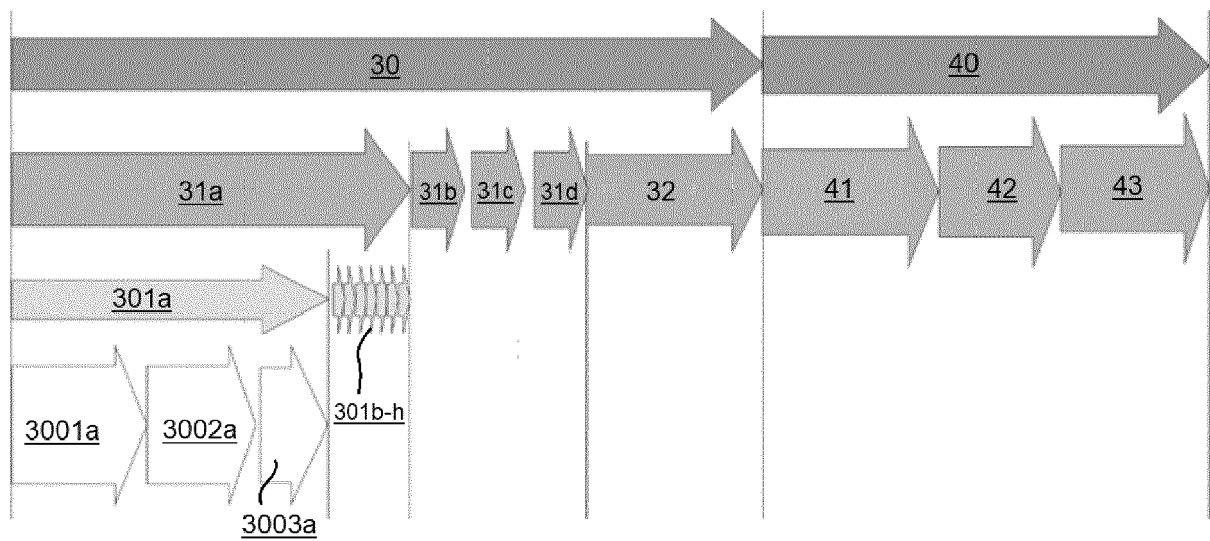


Fig. 3



EUROPEAN SEARCH REPORT

Application Number
EP 21 16 3101

5

10

15

20

25

30

35

40

45

50

55

DOCUMENTS CONSIDERED TO BE RELEVANT			
Category	Citation of document with indication, where appropriate, of relevant passages	Relevant to claim	CLASSIFICATION OF THE APPLICATION (IPC)
X	US 2021/000013 A1 (ROBERTSON DUNCAN [GB] ET AL) 7 January 2021 (2021-01-07) * abstract * * paragraph [0065] * * paragraph [0109] - paragraph [0111] * * paragraphs [0147], [0154] * * paragraphs [0200], [0237] * * paragraphs [0321], [0403] * * paragraphs [0543], [0596] * -----	1-16	INV. G06K9/00 G06K9/62
			TECHNICAL FIELDS SEARCHED (IPC)
			G06K
The present search report has been drawn up for all claims			
Place of search The Hague		Date of completion of the search 7 September 2021	Examiner Peelen, Bastien
CATEGORY OF CITED DOCUMENTS X : particularly relevant if taken alone Y : particularly relevant if combined with another document of the same category A : technological background O : non-written disclosure P : intermediate document T : theory or principle underlying the invention E : earlier patent document, but published on, or after the filing date D : document cited in the application L : document cited for other reasons & : member of the same patent family, corresponding document			

 1
EPO FORM 1503 03.82 (P04C01)

**ANNEX TO THE EUROPEAN SEARCH REPORT
ON EUROPEAN PATENT APPLICATION NO.**

EP 21 16 3101

5

This annex lists the patent family members relating to the patent documents cited in the above-mentioned European search report.
The members are as contained in the European Patent Office EDP file on
The European Patent Office is in no way liable for these particulars which are merely given for the purpose of information.

07-09-2021

10

Patent document cited in search report	Publication date	Patent family member(s)	Publication date
US 2021000013 A1	07-01-2021	AU 2017357645 A1	23-05-2019
		CN 110139552 A	16-08-2019
		EP 3537867 A1	18-09-2019
		US 2019261565 A1	29-08-2019
		US 2019261566 A1	29-08-2019
		US 2021000013 A1	07-01-2021
		WO 2018087546 A1	17-05-2018

15

20

25

30

35

40

45

50

55

EPO FORM P0459

For more details about this annex : see Official Journal of the European Patent Office, No. 12/82

REFERENCES CITED IN THE DESCRIPTION

This list of references cited by the applicant is for the reader's convenience only. It does not form part of the European patent document. Even though great care has been taken in compiling the references, errors or omissions cannot be excluded and the EPO disclaims all liability in this regard.

Patent documents cited in the description

- US 20200008355 A1 [0005]
- CN 109948444 A [0005]
- WO 2020155277 A1 [0005]
- JP 6749720 B [0006]

Non-patent literature cited in the description

- **KRAGIC, D ; CHRISTENSEN, HI.** Survey on visual servoing for manipulation. *Computational Vision and Active Perception Laboratory*, 2002 [0003]
- **QUENTIN BATEUX ; ERIC MARCHAND ; JÜRGEN LEITNER ; FRANÇOIS CHAUMETTE ; PETER CORKE.** Training DeepNeural Networks for Visual Servoing. *IEEE International Conference on Robotics and Automation (ICRA)*, 2018 [0004]
- **ZONGWEI ZHOU ; MD MAHFUZUR RAHMAN SIDDIQUEE ; NIMA TAJBAKSH ; JIANMING LIANG.** UNet++: A Nested U-Net Architecture for Medical Image Segmentation. *4th Deep Learning in Medical Image Analysis (DLMIA) Workshop*, 2018 [0032]
- **YUE WANG ; YONGBIN SUN ; ZIWEI LIU ; SANJAY E. SARMA ; MICHAEL M. BRONSTEIN ; JUSTIN M. SOLOMON.** Dynamic Graph CNN for Learning on Point Clouds. *Computer Vision and Pattern Recognition*, 2019 [0032]
- **HUGUES THOMAS ; CHARLES R. QI ; JEAN-EMMANUEL DESCHAUD ; BEATRIZ MARCOTEGUI ; FRANCOIS GOULETTE ; LEONIDAS J. GUIBAS.** The IEEE International Conference on Computer Vision (ICCV), 2019, 6411-6420 [0032]
- **RONNEBERGER, OLAF ; FISCHER, PHILIPP ; BROX, THOMAS.** U-Net: Convolutional Networks for Biomedical Image Segmentation. *arXiv:1505.04597*, 2015 [0033]
- **CHARLES R. QI et al.** PointNet++: Deep Hierarchical Feature Learning on Point Sets in a Metric Space, 2017, <https://arxiv.org/abs/1706.02413> [0034] [0064]
- **MAURICE WEILER ; GABRIELE CESA.** General E(2)-Equivariant Steerable CNNs. *Conference on Neural Information Processing Systems (NeurIPS)*, 2019 [0056] [0059] [0078]
- **RICHARD ZHANG.** Making Convolutional Networks Shift-Invariant Again. *International Conference on Machine Learning*, 2019 [0059]
- **YUE WANG et al.** Dynamic Graph CNN for Learning on Point Clouds, *CoRR*, 2018, <http://arxiv.org/abs/1801.07829> [0064]